

AMENDMENT

1. (original) A method for intelligently providing application and data in a mobile device system, comprising the steps of:
  - collecting user operation history from at least one user;
  - generating user and application registration information;
  - generating a list of frequently accessed information based on said user operation history;
  - selectively caching information at a mobile device and a gateway based on said list of frequently accessed information, said information including application or data;
  - updating said information at said mobile device and said gateway; and
  - synchronizing said user operation history and said user and application registration information among a plurality of gateways.
2. (original) The method of claim 1, wherein said selectively caching includes the steps of:
  - caching a first set of frequently accessed information of each user at a respective mobile device; and
  - caching a second set of frequently accessed information at said gateway.
3. (original) The method of claim 1, wherein said step of updating said information includes the steps of:
  - receiving a broadcast about a new version of said information;
  - checking a local cache for an old version of said information;
  - comparing said new version and said old version; and
  - differentially updating portions of said information based on said comparing in response to a user event.
4. (original) The method of claim 3, wherein said user event includes a user request to execute or access said old version.
5. (original) The method of claim 1, wherein said step of updating said information includes the steps of:
  - performing a status check with a server;
  - downloading a new version from said server if said new version exists; and
  - replacing an old version with said new version.
6. (original) The method of claim 5, wherein said performing step includes the step of:
  - triggering said status check in response to a user event.
7. (original) The method of claim 5, wherein said performing step includes the step of:
  - automatically triggering said status check in accordance with an estimated update interval.
8. (original) The method of claim 1, wherein said step of updating said information includes the steps of:

BEST AVAILABLE COPY

receiving a broadcast about a new version of said information;  
 checking a local cache at said gateway for an old version of said information;  
 comparing said new version and said old version;  
 generating a differential file based on said comparing; and  
 sending said differential file to said mobile device.

9. (original) The method of claim 1, wherein said step of updating said information includes the steps of:  
 downloading a new version of said information from a server; and  
 checking a local cache at said gateway for an old version of said information:  
     if said local cache includes said old version,  
         comparing said old version and said new version;  
         generating a differential file based on said comparing; and  
         sending said differential file to said mobile device; and  
     if said local cache does not include said old version,  
         sending said new version to said mobile device.
10. (original) The method of claim 9, wherein said downloading step includes the step of:  
 triggering said downloading in response to a schedule.
11. (original) The method of claim 9, wherein said downloading step includes the step of:  
 automatically triggering said downloading in accordance with an estimated update interval.
12. (original) The method of claim 1, wherein said step of synchronizing said information includes the steps of:  
 inquiring a plurality of potential slave gateways about their availability for a synchronization process;  
 receiving responses from said plurality of potential slave gateways;  
 initiating said synchronization process if all of said plurality of potential slave gateways are available; and  
 repeating said inquiring if not all of said plurality of potential slave gateways are available.
13. (original) The method of claim 1, further comprising the steps of:  
 monitoring activity at said mobile device and said gateway; and  
 automatically routing said mobile device to another gateway when said gateway is overloaded, when said gateway crashed, or when a user at said mobile device moved out of an area serviced by said gateway.
14. (original) The method of claim 13, wherein said automatically routing includes the step of:  
 routing said mobile device to a geographically nearest functional gateway.
15. (original) The method of claim 1, further comprising the steps of:  
 saving local information from said mobile device in a temporary cache of said gateway;  
 saving a list of uniform resource locators in a database of said gateway; and

transferring said local information and a copy of said list to a new mobile device upon receiving an initiation request from said new mobile device.

16. (original) The method of claim 15, further comprising the steps of:  
 checking a local cache at said gateway for an application or data corresponding to items on said list of uniform resource locators;  
 if said application or data is found in said local cache:  
     sending said application or data to said new mobile device;  
 if said application or data is not found in said local cache:  
     downloading said application or data from a server; and  
     sending said application or data from said server to said new mobile device.
17. (original) The method of claim 1, further comprising the steps of:  
 receiving a request to search for an application or data, said request including keywords;  
     searching said information at said mobile device and said gateway based on said keywords; and  
     generating a result based on said searching.
18. (original) A method for generating application requests at a mobile device, comprising the steps of:  
     determining whether a requested application is located in a local cache;  
     determining whether said requested application in said local cache is current;  
     determining whether an update schedule is executed for said requested application in said local cache;  
     generating an application download request if said requested application is not located in said local cache;  
     generating an application update request if said requested application in said local cache is not current;  
     generating an application status check request if said update schedule is not executed;  
     opening a logical session;  
     opening or reusing a physical session connected to a gateway; and  
     sending said application download request, said application update request, or said application status check request to said gateway via said logical and said physical sessions.
19. (original) The method of claim 18, further comprising the steps of:  
     searching a user database;  
     collecting user operation history from said user database; and  
     attaching said user operation history to said application download request, said application update request, and said application status check request.
20. (original) The method of claim 18, further comprising the step of:  
     loading said requested application from said local cache if said update schedule is executed.
21. (currently amended) A method for processing requests at a gateway, comprising the steps of:

- parsing a request, said request including user operation history;
  - building a first intelligent strategy based on said parsing, said first intelligent strategy including at least one database access request;
  - accessing a gateway database based on said first intelligent strategy;
  - constructing a response based on said accessing; and
  - sending said response to said a mobile device.
- 22. (original) The method of claim 21, wherein said step of accessing a gateway database includes the steps of:
  - determining whether a requested application or data is located in said gateway;
  - and
  - determining whether said requested application or data is current.
- 23. (original) The method of claim 22, further comprising the steps of:
  - sending said request to a server if said requested application or data is not located in said gateway or said requested application or data is not current;
  - receiving a server response from said server;
  - building a second intelligent strategy based on said server response; and
  - caching or updating said requested application or data in said gateway in accordance with said second intelligent strategy.
- 24. (original) The method of claim 23, wherein said step of sending said request to a server includes the steps of:
  - opening a logical session;
  - opening or reusing a physical session for connection to said server; and
  - sending said request to said server via said logical and physical sessions.
- 25. (original) The method of claim 23, further comprising the steps of:
  - parsing said server response; and
  - sending an acknowledgment to said server if said server response includes any broadcast message.
- 26. (original) A method for providing application and data in a mobile device system, comprising the steps of:
  - sending a schedule for updating an application;
  - periodically updating said application to obtain an updated application;
  - sending a broadcast message about said updated application;
  - receiving a request to update said application;
  - comparing said application to said updated application to obtain a differential file; and
  - sending said differential file in response to said request.
- 27. (original) A computer program product for intelligently providing application and data in a mobile device system, comprising:
  - logic code for collecting user operation history from at least one user;
  - logic code for generating user and application registration information;
  - logic code for generating a list of frequently accessed information based on said user operation history;

logic code for selectively caching information at a mobile device and a gateway based on said list of frequently accessed information, said information including application or data;

logic code for updating said information at said mobile device and said gateway; and

logic code for synchronizing said user operation history and said user and application registration information among a plurality of gateways.

28. (original) The computer program product of claim 27, wherein said logic code for selectively caching includes:

logic code for caching a first set of frequently accessed information of each user at a respective mobile device; and

logic code for caching a second set of frequently accessed information at said gateway.

29. (original) The computer program product of claim 27, wherein said logic code for updating said information includes:

logic code for receiving a broadcast about a new version of said information;

logic code for checking a local cache for an old version of said information;

logic code for comparing said new version and said old version; and

logic code for differentially updating portions of said information based on said comparing in response to a user event.

30. (original) The computer program product of claim 29, wherein said user event includes a user request to execute or access said old version.

31. (original) The computer program product of claim 27, wherein said logic code for updating said information includes:

logic code for performing a status check with a server;

logic code for downloading a new version from said server if said new version exists; and

logic code for replacing an old version with said new version.

32. (original) The computer program product of claim 31, wherein said logic code for performing includes:

logic code for triggering said status check in response to a user event.

33. (original) The computer program product of claim 31, wherein said logic code for performing includes:

logic code for automatically triggering said status check in accordance with an estimated update interval.

34. (original) The computer program product of claim 27, wherein said logic code for updating said information includes:

logic code for receiving a broadcast about a new version of said information;

logic code for checking a local cache at said gateway for an old version of said information;

logic code for comparing said new version and said old version;

logic code for generating a differential file based on said comparing; and

logic code for sending said differential file to said mobile device.

35. (original) The computer program product of claim 27, wherein said logic code for updating said information includes:

logic code for downloading a new version of said information from a server;  
and  
logic code for checking a local cache at said gateway for an old version of said information:

if said local cache includes said old version,  
logic code for comparing said old version and said new version;  
logic code for generating a differential file based on said  
comparing; and  
logic code for sending said differential file to said mobile  
device; and  
if said local cache does not include said old version,  
logic code for sending said new version to said mobile device.

36. (original) The computer program product of claim 35, wherein said logic code for downloading includes:

logic code for triggering said downloading in response to a schedule.

37. (original) The computer program product of claim 35, wherein said logic code for downloading includes:

logic code for automatically triggering said downloading in accordance with an estimated update interval.

38. (original) The computer program product of claim 27, wherein said logic code for synchronizing said information includes:

logic code for inquiring a plurality of potential slave gateways about their availability for a synchronization process;

logic code for receiving responses from said plurality of potential slave gateways;

logic code for initiating said synchronization process if all of said plurality of potential slave gateways are available; and

logic code for repeating said inquiring if not all of said plurality of potential slave gateways are available.

39. (original) The computer program product of claim 27, further comprising:

logic code for monitoring activity at said mobile device and said gateway; and

logic code for automatically routing said mobile device to another gateway when said gateway is overloaded, when said gateway crashed, or when a user at said mobile device moved out of an area serviced by said gateway.

40. (original) The computer program product of claim 39, wherein said logic code for automatically routing includes:

logic code for routing said mobile device to a geographically nearest functional gateway.

41. (original) The computer program product of claim 27, further comprising:

logic code for saving local information from said mobile device in a temporary cache of said gateway;

logic code for saving a list of uniform resource locators in a database of said gateway; and

logic code for transferring said local information and a copy of said list to a new mobile device upon receiving an initiation request from said new mobile device.

42. (original) The computer program product of claim 41, further comprising:
  - logic code for checking a local cache at said gateway for an application or data corresponding to items on said list of uniform resource locators;
  - if said application or data is found in said local cache:
    - logic code for sending said application or data to said new mobile device;
  - if said application or data is not found in said local cache:
    - logic code for downloading said application or data from a server; and
    - logic code for sending said application or data from said server to said new mobile device.
43. (original) The computer program product of claim 27, further comprising:
  - logic code for receiving a request to search for an application or data, said request including keywords;
  - logic code for searching said information at said mobile device and said gateway based on said keywords; and
  - logic code for generating a result based on said searching.
44. (original) A computer program product for generating application requests at a mobile device, comprising:
  - logic code for determining whether a requested application is located in a local cache;
  - logic code for determining whether said requested application in said local cache is current;
  - logic code for determining whether an update schedule is executed for said requested application in said local cache;
  - logic code for generating an application download request if said requested application is not located in said local cache;
  - logic code for generating an application update request if said requested application in said local cache is not current;
  - logic code for generating an application status check request if said update schedule is not executed;
  - logic code for opening a logical session;
  - logic code for opening or reusing a physical session connected to a gateway;

and

  - logic code for sending said application download request, said application update request, or said application status check request to said gateway via said logical and said physical sessions.
45. (original) The computer program product of claim 44, further comprising:
  - logic code for searching a user database;
  - logic code for collecting user operation history from said user database; and
  - logic code for attaching said user operation history to said application download request, said application update request, and said application status check request.

46. (original) The computer program product of claim 44, further comprising:  
 logic code for loading said requested application from said local cache if said update schedule is executed.
47. (original) A computer program product for processing requests at a gateway, comprising:  
 logic code for parsing a request, said request including user operation history;  
 logic code for building a first intelligent strategy based on said parsing, said first intelligent strategy including at least one database access request;  
 logic code for accessing a gateway database based on said first intelligent strategy;  
 logic code for constructing a response based on said accessing; and  
 logic code for sending said response to said mobile device.
48. (original) The computer program product of claim 47, wherein said logic code for accessing a gateway database includes:  
 logic code for determining whether a requested application or data is located in said gateway; and  
 logic code for determining whether said requested application or data is current.
49. (original) The computer program product of claim 48, further comprising:  
 logic code for sending said request to a server if said requested application or data is not located in said gateway or said requested application or data is not current;  
 logic code for receiving a server response from said server;  
 logic code for building a second intelligent strategy based on said server response; and  
 logic code for caching or updating said requested application or data in said gateway in accordance with said second intelligent strategy.
50. (original) The computer program product of claim 49, wherein said logic code for sending said request to a server includes:  
 logic code for opening a logical session;  
 logic code for opening or reusing a physical session for connection to said server; and  
 logic code for sending said request to said server via said logical and physical sessions.
51. (original) The computer program product of claim 49, further comprising:  
 logic code for parsing said server response; and  
 logic code for sending an acknowledgment to said server if said server response includes any broadcast message.
52. (original) A computer program product for providing application and data in a mobile device system, comprising:  
 logic code for sending a schedule for updating an application;  
 logic code for periodically updating said application to obtain an updated application;  
 logic code for sending a broadcast message about said updated application;



logic code for receiving a request to update said application;  
logic code for comparing said application to said updated application to obtain  
a differential file; and  
logic code for sending said differential file in response to said request.